On-the-fly Approximation of
Multivariate Total Variation Minimization

Jordan Frecon*, Nelly Pustelnik*, Patrice Abry*, and Laurent Condat†

Abstract

In the context of change-point detection, addressed by Total Variation minimization strategies, an efficient on-the-fly algorithm has been designed leading to exact solutions for univariate data. In this contribution, an extension of such an on-the-fly strategy to multivariate data is investigated. The proposed algorithm relies on the local validation of the Karush-Kuhn-Tucker conditions on the dual problem. Showing that the non-local nature of the multivariate setting precludes to obtain an exact on-the-fly solution, we devise an on-the-fly algorithm delivering an approximate solution, whose quality is controlled by a practitioner-tunable parameter, acting as a trade-off between quality and computational cost. Performance assessment shows that high quality solutions are obtained on-the-fly while benefiting of computational costs several orders of magnitude lower than standard iterative procedures. The proposed algorithm thus provides practitioners with an efficient multivariate change-point detection on-the-fly procedure.

---

*J. Frecon (Corresponding author), N. Pustelnik and P. Abry are with the Physics Department of the ENS Lyon in France (email: firstname.lastname@ens-lyon.fr).

†L. Condat is with the GIPSA-lab, University of Grenoble, France (email: laurent.condat@gipsa-lab.grenoble-inp.fr).

1 Introduction

Total Variation (TV) has been involved in a variety of signal processing problems, such as nonparametric function estimation [13, 26] or signal denoising [12, 19, 32]. The first contributions on this subject were formulated within the framework of taut string theory [13, 26] while the term TV had first been introduced in image restoration [9, 30]. The equivalence between both formalisms has been clarified in [17].

Formally, the univariate TV framework aims at finding a piece-wise constant estimate $\widehat{\boldsymbol{x}} \in \mathbb{R}^N$ of a noisy univariate signal $\boldsymbol{y} \in \mathbb{R}^N$ by solving the following non-smooth convex optimization problem,

$$\widehat{\boldsymbol{x}} = \underset{\boldsymbol{x}=(x_k)_{1 \le k \le N}}{\arg\min} \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{y}\|^2 + \lambda \sum_{k=1}^{N-1} |x_{k+1} - x_k|, \tag{1}$$

where $\lambda > 0$ denotes a regularization parameter balancing data fidelity versus the piece-wise nature of the solution.

Related works: recent developments and issues. It is well known and documented that the unique solution of the optimization problem (1) can be reached by iterative fixed-point algorithms. On the one hand, solving this problem in the primal space requires to deal with the non-differentiability of the ℓ_1-norm that is either handled by adding a small additional smoothing parameter [33] or by considering proximal algorithms [**?**, 1–3, 5, 10–12, 27, 29, 31]. On the other hand, one can make use of the Fenchel-Rockafellar dual formulation [8, 31] or Lagrangian duality [23, 34] that can be solved with quadratic programming techniques [4, 31]. Both primal and dual solutions suffer from high computational loads, stemming from their iterative nature. To address the computational load issue, alternative procedures were investigated, such as the *taut string algorithm* of common use in the statistics literature [26]. Very recently, elaborating on the dual formulation and thoroughly analysing the related Karush-Kuhn-Tucker (KKT) conditions, a fast algorithm has been proposed by one of the authors in [12] to solve the univariate optimization problem (1). Compared to the taut string strategy, it permits to avoid running sum potentially leading to overflow values and thus numerical errors. Another specificity concerns its *on-the-fly* behavior that does not require the observation of the whole time sequence before a solution can be obtained. On-the-fly algorithms might be of critical interest for real-time monitoring such as in medical applications [7, 15].

Along another line, extension of the univariate optimization problem (1) to multivariate purposes has been recently investigated in [6, 20, 32]. The multivariate extension arises very naturally in numerous contexts, such as biomedical applications, for which the purpose is to extract simultaneous change points from multivariate data, e.g., EEG data [16]. It also encompasses denoising of complex-valued data, which can naturally be interpreted as bivariate data. Multivariate optimization is known as the group fused Lasso in the statistics literature [24, 35]. From a Bayesian point of view, elegant solutions have been proposed in [14, 28] and efficient iterative strategies have recently been proposed in [6, 21].

Mutivariate on-the-fly TV. In this context, the present contribution elaborates on [12] to propose an on-the-fly algorithm solving the multivariate extension of (1). In Section 2, the group fused Lasso problem is first detailed. It is then illustrated that the multivariate procedure has a *non-local* behavior as opposed to the *local* nature of the univariate problem (1). Consequently, any on-the-fly algorithm solving the multivariate minimization problem will only lead to an approximate solution. The KKT conditions resulting from the dual formulation of the multivariate problem are specified in Section 3. From such conditions, a fast and on-the-fly, yet approximate, algorithm is derived in Section 4. The performance in terms of achieved solution and computational gain are presented in Section 5. A video demonstrating the on-the-fly behavior of the algorithm as well as a MATLAB toolbox are available at `http://perso.ens-lyon.fr/jordan.frecon`.

Notations. Let $\mathbf{u} = (u_{m,k})_{1 \leq m \leq M, 1 \leq k \leq N} \in \mathbb{R}^{M \times N}$ denote a multivariate signal, where for every $m \in \{1, \ldots, M\}$, $\boldsymbol{u}_m = (u_{m,k})_{1 \leq k \leq N} \in \mathbb{R}^N$ stands for the m-th component while the k-th values will be shortened as $\mathbf{u}_k = (u_{m,k})_{1 \leq m \leq M} \in \mathbb{R}^M$. For every $k \in \{1, \ldots, N\}$, use will also be made of the following functions: $\mathrm{abs}(\mathbf{u}_k) = (|u_{m,k}|)_{1 \leq m \leq M} \in \mathbb{R}^M$ and $\mathrm{sgn}(\mathbf{u}_k) = (\mathrm{sgn}(u_{m,k}))_{1 \leq m \leq M} \in \mathbb{R}^M$.

2 Local vs non-local Nature

We denote $\mathbf{y}$ the multivariate signal of interest. A multivariate extension of (1) reads:

$$\widehat{\mathbf{x}} = \underset{\mathbf{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M)}{\arg \min} \; \frac{1}{2} \sum_{m=1}^{M} \|\boldsymbol{x}_m - \boldsymbol{y}_m\|^2 + \lambda \sum_{k=1}^{N-1} \sqrt{\sum_{m=1}^{M} |(L\boldsymbol{x}_m)_k|^2}, \tag{2}$$

where $\lambda > 0$ denotes the regularization parameter and $L \in \mathbb{R}^{(N-1) \times N}$ denotes the first order difference operator, that is, for $m \in \{1, \ldots, M\}$ and $k \in \{1, \ldots, N-1\}$,

$$(L\boldsymbol{x}_m)_k = x_{m,k+1} - x_{m,k}. \tag{3}$$

Despite formal similarity, there is however a fundamental difference in nature between the univariate ($M = 1$) and multivariate ($M > 1$) cases: The former is intrinsically *local* [12,25] while the latter is *non-local*[1]. To make explicit such a notion, we have designed the following experiment, whose results are illustrated in Fig. 1. The results associated to the univariate (resp. bivariate) case are presented on the right plots (resp. left plots). A univariate signal $\boldsymbol{y} \in \mathbb{R}^N$ with $N = 180$, consisting of the additive sum of a piece-wise constant signal and white Gaussian noise (in gray, in Fig. 1, right top plot), is considered first. The solution of the minimization problem (1) is displayed in solid red lines in Fig.1. Also, we search for the solution of the minimization problem (1) applied to two partitions of $\boldsymbol{y}$, obtained by splitting it in half, i.e., $\boldsymbol{y}_- = (y_k)_{1 \leq k \leq N/2}$ and $\boldsymbol{y}_+ = (y_k)_{N/2+1 \leq k \leq N}$. The solutions $\boldsymbol{x}_-$ and $\boldsymbol{x}_+$ of (1) respectively associated to $\boldsymbol{y}_-$ and $\boldsymbol{y}_+$ are concatenated and displayed with dashed blue lines in Fig. 1. There is strictly no difference between $\boldsymbol{x}$ and the concatenation of $\boldsymbol{x}_-$ and $\boldsymbol{x}_+$, as reported in Fig.1 (bottom right plot), except for the segment that contains the concatenation point. The difference around the concatenation point is expected as $\boldsymbol{x}$ makes use of an information (the continuity between $\boldsymbol{y}_-$ and $\boldsymbol{y}_+$) that is not available to compute

---

[1]In this article, we denote a problem as local if the solution at a given location does not depend on the signal located earlier (later) than the previous (next) change-point.
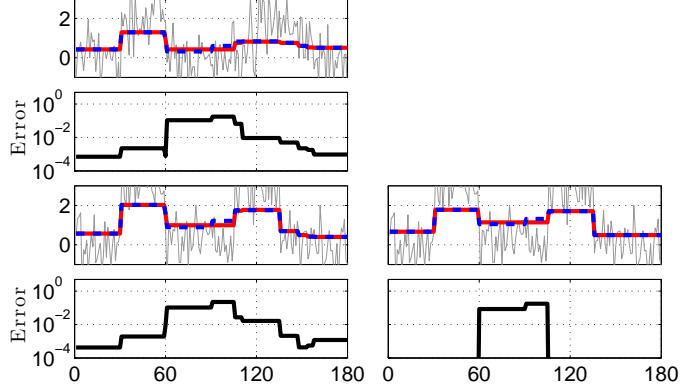
Figure 1: **Non-local vs. local nature.** Left: bivariate TV (upper plots: first component, lower plots: second component). Right: univariate TV. Observations $\mathbf{y}$ (gray), solution $\widehat{\mathbf{x}}$ (red), concatenation of solutions $\widehat{\mathbf{x}}_-$ and $\widehat{\mathbf{x}}_+$ (dashed blue).

$\boldsymbol{x}_-$ and $\boldsymbol{x}_+$. The fact that there is no difference elsewhere shows the local nature of the univariate solution to Problem (1).

This experiment is now repeated for $M = 2$ (as the simplest representative of $M > 1$). A bivariate signal $\mathbf{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2) \in \mathbb{R}^{2 \times N}$ with $N = 180$, consisting of the additive sum of piecewise constant signals and white Gaussian noises (in gray, in Fig. 1, left plots, 1st and 3rd lines), is considered. Two partitions, obtained by splitting in half, $\mathbf{y}_- = (y_{1,k}, y_{2,k})_{1 \leq k \leq N/2}$ and $\mathbf{y}_+ = (y_{1,k}, y_{2,k})_{N/2+1 \leq k \leq N}$ are also considered. The corresponding solutions of (2), applied to $\mathbf{y}, \mathbf{y}_-, \mathbf{y}_+$, labeled $\widehat{\mathbf{x}}$, $\widehat{\mathbf{x}}_-$ and $\widehat{\mathbf{x}}_+$ are obtained by means of the primal-dual algorithm proposed in [**?**] with $\lambda = 20$. Solutions $\widehat{\mathbf{x}}_-$ and $\widehat{\mathbf{x}}_+$ of (2) respectively associated to $\mathbf{y}_-$ and $\mathbf{y}_+$ are concatenated and displayed with dashed blue lines in Fig. 1, while $\widehat{\mathbf{x}}$ is shown in red. Contrary to the case $M = 1$, differences between $\widehat{\mathbf{x}}$ and concatenated $\widehat{\mathbf{x}}_-$ and $\widehat{\mathbf{x}}_+$, shown in black in bottom plots, differ unambiguously from 0 over the entire support of $\mathbf{y}$, clearly showing the *non-local* nature of $\widehat{\mathbf{x}}$ when $M > 1$.

In the univariate case (Eq. (1)), the *local* nature of the solution permits to design an efficient taut string algorithm, that consists in finding the string of minimal length (i.e., taut string) that holds in the tube of radius λ around the antiderivative of $\mathbf{y}$. The solution $\widehat{\boldsymbol{x}}$ of (1) is then obtained by computing the derivative of the taut string. An efficient strategy has been proposed in [13] in order to straightforwardly compute $\widehat{\boldsymbol{x}}$ by determining the points of contact between the taut string and the tube. Even though this approach can be generalized to multivariate signals, the detection of points of contact additionally requires the angle of contact between the taut string and the tube. However, this information is *non-local* and thus the on-the-fly minimization problem results in a challenging contact problem which can not be solved locally. This interpretation will be further discussed in Section 3.

The *non-local* nature of the multivariate ($M > 1$) Problem (2) implies that one cannot expect to find an exact multivariate on-the-fly algorithm. Therefore, in the present work, we will derive an *approximate* on-the-fly algorithm that provides us a good-quality approximation of the exact solution to Problem (2). A control parameter $|\mathcal{Q}|$, defined in Section 3, will control the trade-off between the quality of the approximation and the computational cost.

3 Multivariate Total Variation Minimization

3.1 Dual formulation

Fenchel-Rockafellar dual formulation[2] of (2) reads :

$$\underset{\mathbf{u}\in\mathbb{R}^{M\times(N-1)}}{\text{minimize}} \frac{1}{2}\sum_{m=1}^{M}\|\boldsymbol{y}_m + L^*\boldsymbol{u}_m\|^2 \quad \text{subject to} \quad (\forall k=\{1,\dots,N-1\}) \quad \|\mathbf{u}_k\|\leq\lambda, \quad (4)$$

where, for every $m\in\{1,\dots,M\}$ and $k=\{2,\dots,N-2\}$,

$$(L^*\widehat{\boldsymbol{u}}_m)_k = \widehat{u}_{m,k-1} - \widehat{u}_{m,k} \tag{5}$$

and

$$\begin{cases} (L^*\widehat{\boldsymbol{u}}_m)_1 &= -\widehat{u}_{m,1}, \\ (L^*\widehat{\boldsymbol{u}}_m)_N &= \widehat{u}_{m,N-1}. \end{cases} \tag{6}$$

The optimal solutions $\widehat{\mathbf{u}}\in\mathbb{R}^{M\times(N-1)}$ and $\widehat{\mathbf{x}}\in\mathbb{R}^{M\times N}$ of the dual problem and of the primal problem respectively are related by

$$\begin{cases} (\forall m\in\{1,\dots,M\}) & \widehat{\boldsymbol{x}}_m = \boldsymbol{y}_m + L^*\widehat{\boldsymbol{u}}_m, \\ (\forall k\in\{1,\dots,N-1\}) & \widehat{\mathbf{u}}_k \in -\lambda\partial\|\cdot\|(\widehat{\mathbf{x}}_{k+1}-\widehat{\mathbf{x}}_k). \end{cases} \tag{7}$$

From (7), we directly obtain the following necessary and sufficient conditions.

Proposition 3.1 *The solutions of the primal problem* (2) *and the dual problem* (4) *satisfy, for every* $m\in\{1,\dots,M\}$,

$$\widehat{\boldsymbol{x}}_m = \boldsymbol{y}_m + L^*\widehat{\boldsymbol{u}}_m, \tag{8}$$

and, for every $k=\{1,\dots,N-1\}$,

$$\begin{cases} if \quad \widehat{\mathbf{x}}_k = \widehat{\mathbf{x}}_{k+1} & then \quad \|\widehat{\mathbf{u}}_k\|\leq\lambda, \\ if \quad \widehat{\mathbf{x}}_k \neq \widehat{\mathbf{x}}_{k+1} & then \quad \widehat{\mathbf{u}}_k = -\lambda\frac{\widehat{\mathbf{x}}_{k+1}-\widehat{\mathbf{x}}_k}{\|\widehat{\mathbf{x}}_{k+1}-\widehat{\mathbf{x}}_k\|}. \end{cases} \tag{9}$$

The first condition corresponds to the configuration where every component keeps the same value from location k to $k+1$. This configuration is illustrated in the bivariate case $(M=2)$ in Fig. 2 (left plot). The second condition models situations where some components of $\widehat{\mathbf{x}}$ admit change points between locations k and $k+1$. An interesting configuration is that of non-simultaneous change points as illustrated in Fig. 2 (right plot). In the presence of noise, this second situation is rarely encountered. Thus, in the sequel, we will only consider simultaneous change points.

Remark 3.2 Proposition 3.1 for $M=1$ leads to the usual KKT conditions associated to the minimization problem (1):

$$\begin{cases} if \quad \widehat{x}_k > \widehat{x}_{k+1} & then \quad \widehat{u}_k = +\lambda, \\ if \quad \widehat{x}_k < \widehat{x}_{k+1} & then \quad \widehat{u}_k = -\lambda, \\ if \quad \widehat{x}_k = \widehat{x}_{k+1} & then \quad \widehat{u}_k \in [-\lambda,+\lambda]. \end{cases} \tag{10}$$

The on-the-fly univariate TV algorithm proposed in [12] is derived from Conditions (10).

---

[2]Note that, the usual dual formulation and the resulting stationarity conditions would involve $\mathbf{u}$ rather than $-\mathbf{u}$. The choice made in this article enables us to be consistent with the results obtained in [12] for the univariate case.
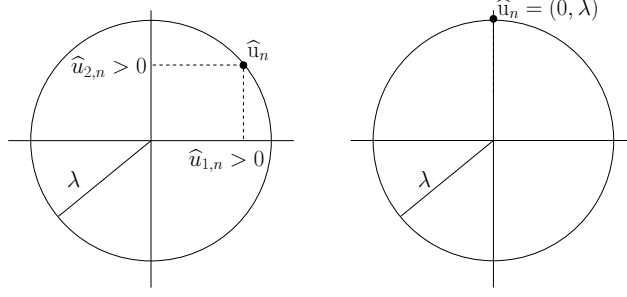
Figure 2: **Comparing joint vs disjoint changes in the dual space.** Left: location k is suitable for a joint negative amplitude change on both components. Right: configuration suitable for introducing a negative amplitude change at k on the second component only.

3.2   Rewriting the KKT conditions

Contrary to Conditions (10), the multivariate conditions derived in Proposition 3.1 are not directly usable in practice to devise an on-the-fly algorithm because $\widehat{\mathbf{x}}_{k+1} - \widehat{\mathbf{x}}_k$ is a priori unknown at instant k. Therefore, we propose to rewrite the second condition in (9) by means of auxiliary variables $(\widehat{\mathbf{z}}_k)_{1 \leq k \leq N-1}$ such that

$$\begin{cases} \text{if} \quad \widehat{\mathbf{x}}_k = \widehat{\mathbf{x}}_{k+1} \quad \text{then} \quad \|\widehat{\mathbf{u}}_k\| \leq \lambda, \\ \text{if} \quad \widehat{\mathbf{x}}_k \neq \widehat{\mathbf{x}}_{k+1} \quad \text{then} \quad \widehat{\mathbf{u}}_k = -\text{sign}(\widehat{\mathbf{x}}_{k+1} - \widehat{\mathbf{x}}_k) \circ \widehat{\mathbf{z}}_k, \end{cases} \tag{11}$$

with $\widehat{\mathbf{z}}_k = \lambda \frac{\text{abs}(\widehat{\mathbf{x}}_{k+1} - \widehat{\mathbf{x}}_k)}{\|\widehat{\mathbf{x}}_{k+1} - \widehat{\mathbf{x}}_k\|}$ and where $\circ$ denotes the Hadamard product. Then, Proposition 3.1, can be reformulated component-wise as follows.

Proposition 3.3 *The solutions of the primal problem* (2) *and of the dual problem* (4) *satisfy the following necessary and sufficient conditions. There exist nonnegative auxiliary variables* $(\widehat{\mathbf{z}}_k)_{1 \leq k \leq N-1}$ *such that, for every* $m = \{1, \dots, M\}$ *and* $k = \{1, \dots, N-1\}$,

$$\begin{cases} \text{if} \quad \widehat{x}_{m,k} > \widehat{x}_{m,k+1} \quad \text{then} \quad \widehat{u}_{m,k} = +\widehat{z}_{m,k}, \\ \text{if} \quad \widehat{x}_{m,k} < \widehat{x}_{m,k+1} \quad \text{then} \quad \widehat{u}_{m,k} = -\widehat{z}_{m,k}, \\ \text{if} \quad \widehat{x}_{m,k} = \widehat{x}_{m,k+1} \quad \text{then} \quad \widehat{u}_{m,k} \in [-\widehat{z}_{m,k}, +\widehat{z}_{m,k}], \end{cases} \tag{12}$$

with $\|\widehat{\mathbf{z}}_k\| = \lambda$ *and* $\widehat{\boldsymbol{x}}_m = \boldsymbol{y}_m + L^*\widehat{\boldsymbol{u}}_m$.

Comparing Eqs. (10) and (12) highlights the similarity between the necessary conditions of the univariate and multivariate minimization problems: Conditions involving λ in the univariate case involve the auxiliary vector $\widehat{\mathbf{z}}$ in the multivariate one. The fact that $\widehat{\mathbf{z}}$ differs for each pair (m, k) can be interpreted in taut string procedures as the fact that the point of contact with the taut string may vary on the tube of radius λ. This significantly increases the difficulty of deriving an on-the-fly algorithm.

3.3 Approximate solution

If we first assume that $\widehat{\mathbf{z}}$ is known and such that, for every $k \in \{1, \ldots, N-1\}$, $\|\widehat{\mathbf{z}}_k\| = \lambda$, the primal problem associated to Conditions (12) reads

$$\min_{\mathbf{x}} \ \sum_{m=1}^{M} \left(\frac{1}{2} \|\boldsymbol{y}_m - \boldsymbol{x}_m\|^2 + \sum_{k=1}^{N-1} \widehat{z}_{m,k} |(L\boldsymbol{x}_m)_k| \right) \tag{13}$$

and can be interpreted as M univariate minimization problems having time-varying regularization parameters $(\widehat{\boldsymbol{z}}_m)_{1 \leq m \leq M}$.

The proposed approximation consists in restricting the estimation of $\widehat{\mathbf{z}}$ to a predefined set $\mathcal{Q} = \{\zeta^{(1)}, \ldots, \zeta^{(|\mathcal{Q}|)}\}$ chosen such as for every $q \in \{1, \ldots, |\mathcal{Q}|\}$, $\zeta^q = (\zeta_m^{(q)})_{1 \leq m \leq M} \in \mathbb{R}^M$ satisfies $\|\zeta^{(q)}\| = \lambda$.

The most naive strategy would consist in solving M univariate minimization problems for every $|\mathcal{Q}|$ candidate values of $\widehat{\mathbf{z}}$, i.e., find for every $m = \{1, \ldots, M\}$ and $q = \{1, \ldots, |\mathcal{Q}|\}$,

$$\widehat{\boldsymbol{x}}_m^{(q)} = \arg\min_{\boldsymbol{x}_m} \frac{1}{2} \|\boldsymbol{y}_m - \boldsymbol{x}_m\|^2 + \zeta_m^{(q)} \|L\boldsymbol{x}_m\|_1 \tag{14}$$

and to devise a method to pick the solution amongst the $|\mathcal{Q}|$ candidates. For instance, the one that maximizes some quality criterion f, i.e.,

$$\widehat{\mathbf{x}} = \widehat{\mathbf{x}}^{(q^*)} \quad \text{with} \quad q^* = \arg\max_{1 \leq q \leq |\mathcal{Q}|} f(\widehat{\mathbf{x}}^{(q)}). \tag{15}$$

Although it benefits from parallel on-the-fly implementations, this situation would correspond to a constant estimate $\tilde{\mathbf{z}} = \zeta^{(q^*)} \mathbb{1}_N$. Therefore, changes in the mean would be processed independently on all components and group-sparsity would not be enforced.

In order to benefit from an on-the-fly implementation and to enforce group-sparsity, we propose an algorithmic solution based on a piece-wise constant estimator of $\widehat{\mathbf{z}}$ detailed in the next section.

4 Algorithmic solution

In the following, we first extend the on-the-fly algorithm proposed in [12] to the multivariate case, with $\widehat{\mathbf{z}}$ assumed to be known a priori. This strong assumption, unrealistic in pratice, permits to describe clearly the behaviour of the multivariate on-the-fly algorithm. Then, we will focus on the question of the automated and on-the-fly estimation of $\widehat{\mathbf{z}}$ taking its values in $\mathcal{Q}$, which consequently introduce a parameter $|\mathcal{Q}|$ controlling the quality of the approximation. The main steps of the on-the-fly algorithm are summarized in Algorithm 1. It is based on the range control of both unknown primal and dual solutions $\widehat{\mathbf{x}}$ and $\widehat{\mathbf{u}}$ by lower and upper bounds updated with the incoming data stream.

The design of Algorithm 1 results in specifying Rule 1 and Rule 2 allowing respectively to detect a change point and to find suitable change-point locations according to Proposition 3.3.

Algorithm 1: On-the-fly scheme for multivariate TV

Data: Multivariate signal $\mathbf{y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M) \in \mathrm{R}^{M \times N}$.
    Regularization parameter $\lambda > 0$.
    Starting location $k_0 = 1$.

while $k_0 < N$ **do**
    Set $k \leftarrow k_0$
    Initialize primal/dual bounds
    while *Rule 1 is satisfied* **do**
        Set $k \leftarrow k + 1$
        for $m \leftarrow 1$ **to** M **do**
            Update primal/dual bounds
            if *Rule 2 is not satisfied* **then**
                Revise the update of primal/dual bounds

    Estimate the change point k_{rupt}
    Estimate $(\widehat{\mathrm{x}}_j)_{k_0 \le j \le k_{\mathrm{rupt}}}$
    Set $k_0 \leftarrow k_{\mathrm{rupt}} + 1$
Result: Solution $\widehat{\mathbf{x}}_{\mathrm{approx}}$

4.1 Ideal case with $\widehat{\mathbf{z}}$ known

4.1.1 Lower and upper bounds

According to Proposition 3.3, the solution of the primal problem, the solution of the dual problem and the auxiliary variable have to satisfy, for every $k \in \{0, \ldots, N-1\}$,

$$\begin{cases} \widehat{\mathbf{u}}_{k+1} = \mathbf{y}_{k+1} + \widehat{\mathbf{u}}_k - \widehat{\mathbf{x}}_{k+1}, \\ \mathrm{abs}(\widehat{\mathbf{u}}_{k+1}) \le \widehat{\mathbf{z}}_{k+1}, \\ \|\widehat{\mathbf{z}}_{k+1}\| = \lambda. \end{cases} \quad (16)$$

with $\widehat{\mathbf{u}}_0 = \widehat{\mathbf{u}}_N = 0$. Considering the two first conditions, the prolongation condition $\widehat{\mathbf{x}}_{k+1} = \widehat{\mathbf{x}}_k$ leads to

$$\begin{cases} \mathbf{y}_{k+1} \ge \widehat{\mathbf{x}}_k - \widehat{\mathbf{z}}_{k+1} - \widehat{\mathbf{u}}_k, \\ \mathbf{y}_{k+1} \le \widehat{\mathbf{x}}_k + \widehat{\mathbf{z}}_{k+1} - \widehat{\mathbf{u}}_k. \end{cases} \quad (17)$$

Following the solution proposed for the univariate case derived in [12], one can check that (17) is satisfied by reasoning on lower and upper bounds of $\widehat{\mathbf{u}}_k$ and $\widehat{\mathbf{x}}_k$. For every $k \in \{1, \ldots, N-1\}$, we define the lower and upper bounds of $\widehat{\mathbf{x}}_k$, labeled $\underline{\mathbf{x}}_k$ and $\overline{\mathbf{x}}_k$ respectively, as:

$$\underline{\mathbf{x}}_k \le \widehat{\mathbf{x}}_k \le \overline{\mathbf{x}}_k, \quad (18)$$

and we set $\underline{\mathbf{u}}_k$ and $\overline{\mathbf{u}}_k$ as follows

$$(\forall m \in \{1, \ldots, M\}) \quad \begin{cases} \widehat{u}_{m,k} = \underline{u}_{m,k} & \text{if} \quad \widehat{x}_{m,k} = \underline{x}_{m,k}, \\ \widehat{u}_{m,k} = \overline{u}_{m,k} & \text{if} \quad \widehat{x}_{m,k} = \overline{x}_{m,k}, \end{cases} \quad (19)$$

where $\underline{\mathbf{u}}_k$ and $\overline{\mathbf{u}}_k$ appear to be the upper and lower bounds of $\widehat{\mathbf{x}}_k$ respectively, i.e.

$$\overline{\mathbf{u}}_k \le \widehat{\mathbf{u}}_k \le \underline{\mathbf{u}}_k, \quad (20)$$

as detailed in Appendix 7.1.

4.1.2 Updating rules & Rule 1

The prolongation condition $\widehat{\mathbf{x}}_{k+1} = \widehat{\mathbf{x}}_k$, which has led to (17), becomes

$$\begin{cases} \mathbf{y}_{k+1} \geq \underline{\mathbf{x}}_k - \widehat{\mathbf{z}}_{k+1} - \underline{\mathbf{u}}_k, \\ \mathbf{y}_{k+1} \leq \overline{\mathbf{x}}_k + \widehat{\mathbf{z}}_{k+1} - \overline{\mathbf{u}}_k. \end{cases} \tag{21}$$

If the latter condition, labeled as Rule 1, holds, then according to the primal-dual relation, we perform the update of the lower and upper bounds at location $k+1$ as follows:

$$\begin{cases} \underline{\mathbf{u}}_{k+1} = \mathbf{y}_{k+1} + \underline{\mathbf{u}}_k - \underline{\mathbf{x}}_k, \\ \overline{\mathbf{u}}_{k+1} = \mathbf{y}_{k+1} + \overline{\mathbf{u}}_k - \overline{\mathbf{x}}_k, \end{cases} \tag{22}$$

and

$$\begin{cases} \underline{\mathbf{x}}_{k+1} = \underline{\mathbf{x}}_k, \\ \overline{\mathbf{x}}_{k+1} = \overline{\mathbf{x}}_k. \end{cases} \tag{23}$$

Remark 4.1 Equivalently, one can systematically update primal (resp. dual) bounds according to (22) (resp. (23)) and verify that the following rewriting of the prolongation condition (21) holds:

$$\begin{cases} \underline{\mathbf{u}}_{k+1} \geq -\widehat{\mathbf{z}}_{k+1}, \\ \overline{\mathbf{u}}_{k+1} \leq +\widehat{\mathbf{z}}_{k+1}. \end{cases} \tag{24}$$

4.1.3 Signal prolongation & Rule 2

If Rule 1 (i.e. Condition (21) or equivalently (24)) holds, then the assumption $\widehat{\mathbf{x}}_{k+1} = \widehat{\mathbf{x}}_k$ is valid. However, the upper and lower bounds may have to be updated in order to be consistent with $\widehat{\mathbf{u}}_{k+1} \in [-\widehat{\mathbf{z}}_{k+1}, +\widehat{\mathbf{z}}_{k+1}]$. According to (20), this condition requires to verify that the following Rule 2 holds:

$$\begin{cases} \underline{\mathbf{u}}_{k+1} \leq +\widehat{\mathbf{z}}_{k+1}, \\ \overline{\mathbf{u}}_{k+1} \geq -\widehat{\mathbf{z}}_{k+1}. \end{cases} \tag{25}$$

For every $m \in \{1, \ldots, M\}$, three configurations can be encountered:

- When both Conditions (25) are satisfied, the bounds are left unchanged.

- When $\underline{u}_{m,k+1} = \underline{u}_{m,k} + y_{m,k+1} - \underline{x}_{m,k} > +\widehat{z}_{m,k+1}$, then the updating rules specified in (23) have under-evaluated the bound

$$\underline{\nu}_m \equiv \underline{x}_{m,j} \quad (\forall j \in \{k_0, \ldots, k+1\}) \tag{26}$$

  where k_0 denotes the last starting location of a new segment. Since $\underline{u}_{m,k+1}$ is upper-bounded by $+\widehat{z}_{m,k+1}$ and, that for such a value it can be shown (see Appendix 7.2) that

$$\underline{\nu}_m = \underline{x}_{m,k} + \frac{\underline{u}_{m,k+1} - \widehat{z}_{m,k+1}}{k - k_0 + 1}, \tag{27}$$

we propose the following updates

$$\begin{cases} (\forall j \in \{k_0, \ldots, k+1\}) \quad \underline{x}_{m,j} = \underline{\nu}_m, \\ \underline{u}_{m,k+1} = +\widehat{z}_{m,k+1}. \end{cases} \tag{28}$$

• When $\overline{u}_{m,k+1} < -\widehat{z}_{m,k+1}$, then it results that the upper bound

$$\overline{\nu}_m \equiv \overline{x}_{m,j} \quad (\forall j \in \{k_0, \ldots, k+1\}) \tag{29}$$

has been over-evaluated. Similarly, since $\overline{u}_{m,k+1}$ is lower bounded by $-\widehat{z}_{m,k+1}$, we can show that the upper bound

$$\overline{\nu}_m = \overline{x}_{m,k} + \frac{\overline{u}_{m,k+1} + \widehat{z}_{m,k+1}}{k - k_0 + 1}, \tag{30}$$

permits to ensure the consistency of the following updates

$$\begin{cases} (\forall j \in \{k_0, \ldots, k+1\}) \quad \overline{x}_{m,j} = \overline{\nu}_m, \\ \overline{u}_{m,k+1} = -\widehat{z}_{m,k+1}. \end{cases} \tag{31}$$

4.1.4 Estimate of the change point k_{rupt}

When Rule 1 does not hold, a change point has to be created. For every $m \in \{1, \ldots, M\}$, we can distinguish three cases:

• When $\underline{u}_{m,k+1} = \underline{u}_{m,k} + y_{m,k+1} - \underline{x}_{m,k} < -\widehat{z}_{m,k+1}$, then, since $\underline{u}_{m,k}$ is bounded, it means that $\underline{x}_{m,k}$ is over-evaluated and therefore a negative amplitude change has to be introduced on the m-th component in the time index set $\{k_0, \ldots, k\}$ in order to decrease its value. Following Proposition 3.3 and Eq. (20), the set of locations κ_m suitable for a change-point on the m-th component reads:

$$\kappa_m = \{j \in \{k_0, \ldots, k\} \quad | \quad \underline{u}_{m,j} = +\widehat{z}_{m,j}\}. \tag{32}$$

Such locations correspond to the indexes where the value of the bound $\underline{u}_{m,j}$ has been updated in order to be consistent with the condition $\widehat{u}_{m,j} \in [-\widehat{z}_{m,j}, \widehat{z}_{m,j}]$ (see the previous paragraph)

• When $\overline{u}_{m,k+1} > +\widehat{z}_{m,k+1}$, then a positive amplitude change has to be introduced in the m-th component within the time index $\{k_0, \ldots, k\}$. The set of locations suitable for a change-point on the m-th component reads:

$$\kappa_m = \{j \in \{k_0, \ldots, k\} \quad | \quad \overline{u}_{m,j} = -\widehat{z}_{m,j}\}. \tag{33}$$

This set of locations corresponds to indexes where the value of the bound $\overline{u}_{m,j}$ was updated in order to be consistent with $\widehat{u}_{m,j} \in [-\widehat{z}_{m,j}, \widehat{z}_{m,j}]$.

• Else, when component m does satisfy (17), then we set $\kappa_m = \{k_0, \ldots, k\}$.

The change-point location k_{rupt} corresponds to the last location suitable for introducing the adequate amplitude change on each component, i.e.,

$$k_{\text{rupt}} = \max_{j \in \cap_{m=1}^{M} \kappa_m} j. \tag{34}$$

Once the change point location has been specified, we are able to assign a value to $(\widehat{\mathbf{x}}_j)_{k_0 \leq j \leq k_{\text{rupt}}}$. When a negative amplitude change is detected on the m-th component, we set

$$(\forall j \in \{k_0, \ldots, k_{\text{rupt}}\}) \quad \widehat{x}_{m,j} = \underline{x}_{m,k+1}, \tag{35}$$

in consistence with (19). Similarly, when a positive amplitude change is detected, we set

$$(\forall j \in \{k_0, \ldots, k_{\text{rupt}}\}) \quad \widehat{x}_{m,j} = \overline{x}_{m,k+1}. \tag{36}$$

4.1.5 Starting a new segment

When a segment has been created, we start the detection of a new segment considering $k_0 = k_{\text{rupt}} + 1$ as long as $k_0 < N$.

According to (7) and by definition of the bounds, for every $k \in \{1, \ldots, N\}$

$$\begin{cases} \underline{\mathbf{x}}_k = \mathbf{y}_k - \underline{\mathbf{u}}_k + \widehat{\mathbf{u}}_{k-1}, \\ \overline{\mathbf{x}}_k = \mathbf{y}_k - \overline{\mathbf{u}}_k + \widehat{\mathbf{u}}_{k-1}. \end{cases} \tag{37}$$

In particular, for $k = k_0$, combining (12), (18), (19) and (22) allows us to find the following initialization procedure

$$\begin{cases} \underline{\mathbf{u}}_{k_0} = +\widehat{\mathbf{z}}_{k_0}, \\ \overline{\mathbf{u}}_{k_0} = -\widehat{\mathbf{z}}_{k_0}, \\ \underline{\mathbf{x}}_{k_0} = \mathbf{y}_{k_0} - \widehat{\mathbf{z}}_{k_0} + \widehat{\mathbf{u}}_{k_0-1}, \\ \overline{\mathbf{x}}_{k_0} = \mathbf{y}_{k_0} + \widehat{\mathbf{z}}_{k_0} + \widehat{\mathbf{u}}_{k_0-1}, \end{cases} \tag{38}$$

where the value of $\widehat{\mathbf{u}}_{k_0-1}$ is given according to Proposition 3.3. In addition, according to the writing of (16), $\widehat{\mathbf{u}}_0 = 0$.

4.2 Estimation of the auxiliary multivariate vector $\widehat{\mathbf{z}}$

In order to describe the generic behavior of the multivariate on-the-fly algorithm, we have so far assumed $\widehat{\mathbf{z}}$ to be known a priori. We now focus on the simultaneous estimation of the multivariate vector $\widehat{\mathbf{z}}$ and of the multivariate signal $\widehat{\mathbf{x}}$.

To provide an on-the-fly approximate solution, we propose:

- to build a piece-wise constant estimator $\widetilde{\mathbf{z}}$ of $\widehat{\mathbf{z}}$,

- to only consider amplitude changes jointly on all components $m \in \{1, \ldots, M\}$.

11

4.2.1 Piece-wise constant estimator of $\widehat{\mathbf{z}}$

The proposed estimate is assumed to be constant between each change-point with values belonging to the predefined set $\mathcal{Q}$ defined in Section 3.3. For each candidate value $\zeta^{(q)}$ with $q \in \{1, \ldots, |\mathcal{Q}|\}$, we create upper and lower bounds labeled $\underline{\mathbf{u}}_k^{(q)}$, $\overline{\mathbf{u}}_k^{(q)}$, $\underline{\mathbf{x}}_k^{(q)}$, and $\overline{\mathbf{x}}_k^{(q)}$. They are initialized at each new segment location k_0 and are updated independently according to (22) and (23) until the prolongation condition

$$\begin{cases} \underline{\mathbf{u}}_{k+1}^{(q)} \geq -\zeta^{(q)}, \\ \overline{\mathbf{u}}_{k+1}^{(q)} \leq +\zeta^{(q)}, \end{cases} \tag{39}$$

based on (24), does not hold anymore. In the following, we investigate how to modify the algorithm described in Section 4.1, to account for the automated selection of $\widetilde{\mathbf{z}}$ in $\mathcal{Q}$. The resulting algorithm is reported in Algorithm 2.

4.2.2 Estimate of the change point $k_{\mathrm{rupt}}^{(q)}$

For every $q \in \{1, \ldots, |\mathcal{Q}|\}$, we create change points as described in Section 4.1.4. The main difference consists in the restriction to simultaneous change points. As detailed after Proposition 3.1, non-simultaneous changes have a zero probability to occur. The restriction to simultaneous change-points will thus not impact the solution. It results that if there exists at least one component $m \in \{1, \ldots, M\}$ such that $\underline{u}_{m,k+1}^{(q)} < -\zeta_m^{(q)}$ (resp. $\overline{u}_{m,k+1}^{(q)} > \zeta_m^{(q)}$), then

$$\kappa_m^{(q)} = \{j \in \{k_0, \ldots, k\} \mid \underline{u}_{m,j}^{(q)} = +\zeta_m^{(q)}\} \tag{40}$$

$$(\text{resp.} \quad \kappa_m^{(q)} = \{j \in \{k_0, \ldots, k\} \mid \overline{u}_{m,j}^{(q)} = -\zeta_m^{(q)}\}), \tag{41}$$

and, $\forall m_- \neq m$ such that $\underline{u}_{m_-,k+1}^{(q)} + \overline{u}_{m_-,k+1}^{(q)} < 0$, then

$$\kappa_{m_-}^{(q)} = \{j \in \{k_0, \ldots, k\} \mid \underline{u}_{m_-,j}^{(q)} = +\zeta_m^{(q)}\} \tag{42}$$

or, $\forall m_+ \neq m$ such that $\underline{u}_{m_+,k+1}^{(q)} + \overline{u}_{m_+,k+1}^{(q)} \geq 0$, then

$$\kappa_{m_+}^{(q)} = \{j \in \{k_0, \ldots, k\} \mid \underline{u}_{m_+,j}^{(q)} = +\zeta_m^{(q)}\}. \tag{43}$$

A bivariate example of these configurations where the second component breaks Condition (39) is provided in Fig. 3. The change-point location $k_{\mathrm{rupt}}^{(q)}$ and the assignment of $\widehat{\mathbf{x}}^{(q)}$ on the current segment follow (34), (35) and (36).

4.2.3 Estimate of the change point k_{rupt}

According to the previous paragraph, the piece-wise estimation procedure leads to several possible change-point locations (at most $|\mathcal{Q}|$). Here we select the solution indexed by q^* with tightest bounds $\underline{\mathbf{x}}^{(q^*)}$ and $\overline{\mathbf{x}}^{(q^*)}$, i.e.,

$$q^* \in \underset{1 \leq q \leq |\mathcal{Q}|}{\mathrm{Argmin}} \left\| \left(\overline{\mathbf{x}}_{k_{\mathrm{rupt}}^{(q)}}^{(q)} - \underline{\mathbf{x}}_{k_{\mathrm{rupt}}^{(q)}}^{(q)} \right) \boldsymbol{\sigma}^{-1} \right\|^2, \tag{44}$$
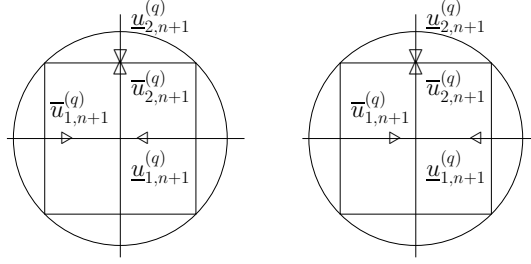
12

Figure 3: **Example of configurations leading to the detection of a change-point**. In this example $M = 2$, $\zeta_1^{(q)} = \zeta_2^{(q)} = \lambda/\sqrt{2}$. Since $\underline{u}_{2,k+1}^{(q)} > \zeta_2^{(q)}$, condition (39) is violated. The left (resp. right) plot displays the configuration $\overline{u}_{1,k+1}^{(q)} + \underline{u}_{1,k+1}^{(q)} < 0$ (resp. $\overline{u}_{1,k+1}^{(q)} + \underline{u}_{1,k+1}^{(q)} \geq 0$) described in Section 4.2.2.

with

$$\boldsymbol{\sigma} = \mathrm{diag}(\sigma_1, \ldots, \sigma_M), \tag{45}$$

where, for every $m \in \{1, \ldots, M\}$, σ_m stands for the standard deviation of $\boldsymbol{y}_m$. The factor $\boldsymbol{\sigma}^{-1}$ permits to ensure that every component contributes equally to the criterion (44). When the minimizer of (44) is not unique, we select the index q^* yielding the largest $k_{\mathrm{rupt}}^{(q^*)}$. In other words, we choose the set of auxiliary variables which permits to hold the prolongation condition (39) as long as possible.

Therefore, it finally leads to an index q^* which permits to estimate $k_{\mathrm{rupt}} = k_{\mathrm{rupt}}^{(q^*)}$ and,

$$(\forall j \in \{k_0, \ldots, k_{\mathrm{rupt}}\}) \quad \tilde{\mathbf{z}}_j = \zeta^{(q^*)}, \ \widehat{\mathbf{x}}_j = \widehat{\mathbf{x}}_j^{(q^*)}. \tag{46}$$

The starting location for the next segment is then, $k_0 = k_{\mathrm{rupt}} + 1$, and the algorithm iterates as long as $k_0 < N$.

4.2.4 Starting a new segment

Let us consider the location k_0 of a new segment. For every $q \in \{1, \ldots, |\mathcal{Q}|\}$, the *initialization* step can be recast into

$$\begin{cases} \underline{\mathbf{u}}_{k_0}^{(q)} = +\zeta^{(q)}, & \overline{\mathbf{u}}_{k_0}^{(q)} = -\zeta^{(q)}, \\ \underline{\mathbf{x}}_{k_0}^{(q)} = \mathbf{y}_{k_0} - \zeta^{(q)} + \widehat{\mathbf{u}}_{k_0-1}, & \overline{\mathbf{x}}_{k_0}^{(q)} = \mathbf{y}_{k_0} + \zeta^{(q)} + \widehat{\mathbf{u}}_{k_0-1}, \end{cases} \tag{47}$$

with $\widehat{\mathbf{u}}_0 = 0$.

Remark 4.2 The initialization step (47) implicitly depends on the estimation of $\widehat{\mathbf{z}}$ made on the last segment through the term $\widehat{\mathbf{u}}_{k_0-1}$. Simulations have shown that (47) may lead to an inconsistent solution $\widehat{\mathbf{x}}$ as soon as $\widehat{\mathbf{z}}$ has been poorly estimated on a segment. Empirically, a better approximation of the iterative solution is observed if each segment is treated independently, i.e.,

$$\begin{cases} \underline{\mathbf{u}}_{k_0}^{(q)} = +\zeta^{(q)}, & \overline{\mathbf{u}}_{k_0}^{(q)} = -\zeta^{(q)}, \\ \underline{\mathbf{x}}_{k_0}^{(q)} = \mathbf{y}_{k_0} - \zeta^{(q)}, & \overline{\mathbf{x}}_{k_0}^{(q)} = \mathbf{y}_{k_0} + \zeta^{(q)}. \end{cases} \tag{48}$$

Algorithm 2: On-the-fly Multivariate TV

Data: Multivariate signal $\mathbf{y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M) \in \mathbb{R}^{M \times N}$.

Regularization parameter $\lambda > 0$.

Predefined set $\mathcal{Q} = \{\zeta^{(1)}, \ldots, \zeta^{(|\mathcal{Q}|)}\}$.

Starting location $k_0 = 1$.

while $k_0 < N$ **do**

    for $q \leftarrow 1$ **to** $|\mathcal{Q}|$ **do**

        Set $k \leftarrow k_0$

        Initialize primal/dual bounds according to (48)

        while (39) *is satisfied* **do**

            Set $k \leftarrow k + 1$

            for $m \leftarrow 1$ **to** M **do**

                Update primal/dual bounds

                if $\underline{u}_{m,k+1}^{(q)} > +\zeta_m^{(q)}$ *or* $\overline{u}_{m,k+1}^{(q)} < -\zeta_m^{(q)}$ **then**

                    Revise the update of primal/dual bounds

        Estimate $k_{\mathrm{rupt}}^{(q)}$ and $(\widehat{\mathbf{x}}_j^{(q)})_{k_0 \leq j \leq k_{\mathrm{rupt}}^{(q)}}$ according to Section (4.2.2)

    Estimate $k_{\mathrm{rupt}} \in (k_{\mathrm{rupt}}^{(q)})_{1 \leq q \leq |\mathcal{Q}|}$ according to Section 4.2.3

    Estimate $(\widehat{\mathbf{x}}_j)_{k_0 \leq j \leq k_{\mathrm{rupt}}}$ and $(\tilde{\mathbf{z}}_j)_{k_0 \leq j \leq k_{\mathrm{rupt}}}$ according to (46)

    Set $k_0 \leftarrow k_{\mathrm{rupt}} + 1$

Result: Solution $\widehat{\mathbf{x}}$

---

5 Performance assessment

5.1 Experimental setting

Unless specified otherwise, we consider that data consist of a M-multivariate piece-wise constant signal $\mathbf{x} \in \mathbb{R}^N$ (solid black), to which a centered Gaussian noise $\boldsymbol{\epsilon}$ is additively superimposed: $\mathbf{y} = \mathbf{x} + \boldsymbol{\epsilon} \in \mathbb{R}^{M \times N}$.

Signal $\mathbf{x}$ is generated as follows. First the length of each segment is drawn according to a folded Gaussian distribution $\mathcal{N}(12.5, 16.25)$. Then, for each $m \in \{1, \ldots, M\}$, the amplitudes of the corresponding changes are drawn independently from a Gaussian distribution $\mathcal{N}(2, 0.4)$.

The exact minimizer of (2), labeled $\widehat{\mathbf{x}}$, is computed by means of the ADMM algorithm proposed in [34]. Iterations are stopped when the relative criterion error is lower than 10^{-10}. The proposed solution computed with the predefined set $\mathcal{Q}$ is denoted $\widehat{\mathbf{x}}_{\mathrm{approx}, \mathcal{Q}}$.

In a second set of experiments (see 5.4), the proposed on-the-fly algorithmic solution will be compared to an on-the-fly ADMM solution.

5.2 Design of $\mathcal{Q}$

We propose to compare solutions $\widehat{\mathbf{x}}_{\mathrm{approx}, \mathcal{Q}}$ obtained with two sets $\mathcal{Q} = \{\zeta^{(1)}, \ldots, \zeta^{(|\mathcal{Q}|)}\}$ in the bivariate case (i.e., $M = 2$) for $N = 10^4$. For both configurations, we choose

$$(\forall q \in \{1, \ldots, |\mathcal{Q}|\}) \quad \zeta^{(q)} = (\lambda \cos(\theta_q), \lambda \sin(\theta_q)) \tag{49}$$

with $\quad \theta_q \in [0, \pi/2]$. The first solution consists to homogeneously cover the ℓ_2 ball such that, for some some parameter $R \in \mathbb{N}_*$, $\theta_q = q\pi/2^{R+1}$ and $|\mathcal{Q}| = \sum_{q'=0}^{R-1} 2^{q'}$. The second
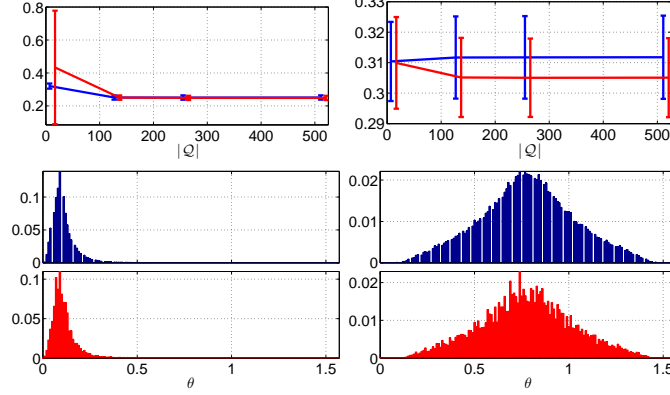
Figure 4: **Influence of the design of $\mathcal{Q}$.** Comparison over 100 realizations of an homogeneous covering of the ℓ_2-ball (blue) against a random covering (red). Two experimental settings are considered depending on whether if $\boldsymbol{y}_1$ is one order of magnitude larger than $\boldsymbol{y}_2$ (left) or not (right). Top: $\mathrm{MSE}(\widehat{\mathbf{x}}_{\mathrm{approx},\mathcal{Q}}, \widehat{\mathbf{x}})$ for different set sizes $|\mathcal{Q}|$. Bottom (2nd and 3rd lines): distributions of θ_{q^*} where q^* has been selected by criterion (44) for $|\mathcal{Q}| = 127$.

solution draws a set of the same size whose values $(\theta_q)_{1 \leq q \leq |\mathcal{Q}|}$ follow a uniform distribution on $[0, \pi/2]$.

Two experimental settings are investigated. In the first one, $\boldsymbol{y}_1$ is one order of magnitude larger than $\boldsymbol{y}_2$ (Fig. 4, left plots) whereas in the second one, both are of the same order of magnitude (Fig. 4, right plots).

Estimation performances in terms of mean squared error $\mathrm{MSE}(\widehat{\mathbf{x}}_{\mathrm{approx},\mathcal{Q}}, \widehat{\mathbf{x}}) = \widehat{\mathbb{E}}[\frac{1}{N}\|\widehat{\mathbf{x}}_{\mathrm{approx},\mathcal{Q}} - \widehat{\mathbf{x}}\|^2]$ (where $\widehat{\mathbb{E}}$ stands for the sample mean estimator computed over 100 realizations) are reported on the first line. It shows that a random covering of the ℓ_2-ball provides solutions as good as the homogeneous covering up to the limit of $|\mathcal{Q}|$ small.

On the 2nd and 3rd lines, the distributions of θ_{q^*}, where q^* has been selected by criterion (44), are reported for $|\mathcal{Q}| = 127$. These histograms show the impact of the relative amplitude of the components on the distribution θ_{q^*}: components with same order of magnitude yield a symmetric distribution while unbalanced components yield an asymmetric distribution. For instance, in Fig. 4 (right plots), it appears more meaningful to draw θ_q according to a Gaussian distribution than to a uniform distribution. Therefore, if one has some knowledge of components amplitudes, this can be incorporated to better design the set $\mathcal{Q}$. This will also decrease the computational cost discussed in section 5.4.

In the following, we restrict ourselves to a random covering of the ℓ_2 ball.

5.3   Offline performance

In this section, we focus on the comparison of offline performance, extended for $M = 10$, for two different signal-to-noise ratios (SNRs), namely 4dB and 10dB.

Qualitative impact of $|\mathcal{Q}|$ on $\widehat{\mathbf{x}}_{\mathrm{approx},\mathcal{Q}}$. For a single realization of noise, $\widehat{\mathbf{x}}_{\mathrm{approx},\mathcal{Q}}$ and $\widehat{\mathbf{x}}$ are plotted Fig. 5 for $\lambda = 29$, adjusted to provide the best visual (qualitative) performance. Solution $\widehat{\mathbf{x}}_{\mathrm{approx},\mathcal{Q}}$ for $|\mathcal{Q}| = 5 \times 10^4$ (light orange) provides a visually better approximation of $\widehat{\mathbf{x}}$ (dashed blue) than for $|\mathcal{Q}| = 10^3$ (mixed red).
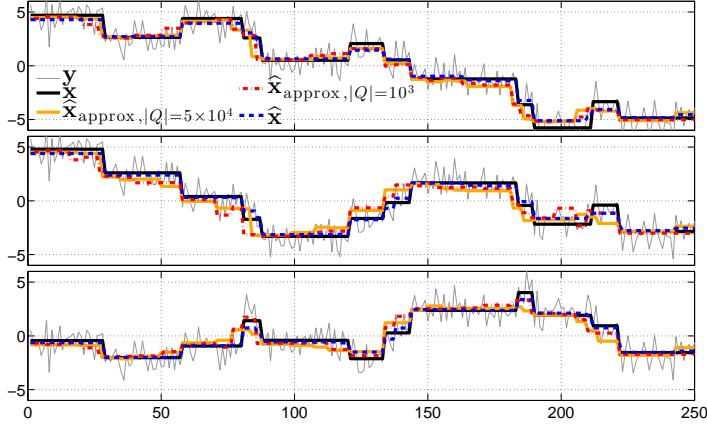
15

Figure 5: **Qualitative impact of** $|\mathcal{Q}|$ **on** $\widehat{\mathbf{x}}_{\mathbf{approx},\mathcal{Q}}$**.** For visibility, only 3 components out of $M = 10$ are displayed. SNR = 4dB. $\widehat{\mathbf{x}}_{\mathbf{approx},\mathcal{Q}}$ for $|\mathcal{Q}| = 5 \times 10^4$ is more satisfying than for $|\mathcal{Q}| = 10^3$ since it has more discontinuities in common with $\widehat{\mathbf{x}}$.

Estimation performance $\widehat{\mathbf{x}}_{\mathbf{approx},\mathcal{Q}}$ **vs.** $\widehat{\mathbf{x}}$**.** The quality of the approximation is further quantified Fig. 6 in terms of $\mathrm{MSE}(\widehat{\mathbf{x}}_{\mathbf{approx},\mathcal{Q}}, \widehat{\mathbf{x}})$ as a function of λ for different $|\mathcal{Q}|$.

It shows that the MSE systematically decreases when $|\mathcal{Q}|$ increases. Further, on the examples considered here and depending on λ, using $|\mathcal{Q}| \geq 10^4$ no longer yields significantly improved solutions, thus showing that the selection of $|\mathcal{Q}|$ does not require a complicated tuning procedure.

Estimation performance $\widehat{\mathbf{x}}$ **vs.** $\mathbf{x}$ **and** $\widehat{\mathbf{x}}_{\mathbf{approx},Q}$ **vs.** $\mathbf{x}$**.** Let us now compare the absolute quality of the solutions against $\mathbf{x}$.

$\mathrm{MSE}(\widehat{\mathbf{x}}, \mathbf{x})$ and $\mathrm{MSE}(\widehat{\mathbf{x}}_{\mathbf{approx},\mathcal{Q}}, \mathbf{x})$ for different $|\mathcal{Q}|$, are reported in Fig. 7. MSEs are consistent with the previous paragraph: it shows that increasing $|\mathcal{Q}|$ up to a certain value permits to significantly lower the MSE. However, $\widehat{\mathbf{x}}$ has a lower estimation error than $\widehat{\mathbf{x}}_{\mathbf{approx},Q}$.

5.4 Online performance

In this section we focus on the comparison between two online solutions. The first one is derived from the proposed on-the-fly algorithm whereas the second one is based on an iterative algorithm.

Comparison is made for different values of λ on a signal $\mathbf{x} \in \mathbb{R}^{M \times N}$ ($N = 400$) to which a Gaussian noise is superimposed such that SNR = 3dB. Performance are provided for $M = 2$ and $M = 5$ components.

Proposed online solution $\widehat{\mathbf{x}}_{\mathbf{online},\mathcal{Q}}$**.** As the time step k increases, $\widehat{\mathbf{x}}_{\mathbf{approx},\mathcal{Q}}$ is only computed up to the last k_0 and the algorithm has not yet output a solution on $\{k_0 + 1, \ldots, k\}$. In that sense, the solution is said to be "on-the-fly". However, a solution $\widehat{\mathbf{x}}_{\mathbf{online},\mathcal{Q}}$, providing an online approximation of $\mathbf{x}$, can be output up to k by imposing limit conditions at k.

Windowed iterative solution $\widehat{\mathbf{x}}_{\text{win},K}$. We consider a naive online ADMM version, where at each time step k a solution $\widehat{\mathbf{x}}_{\text{win},K}$ is computed by optimizing over the previous K points. The choice of K is of critical importance. On the one hand, if this value is too small, the observer may miss amplitude changes in the multivariate data stream. On the other hand, if the window size is too large, the computational cost may be too high to handle any online observation. Three window sizes have been investigated, respectively $K = 20, 50$ and 80.

Computational cost. Comparisons of median computational costs per incoming sample (in seconds), over 10 realizations of noise, are reported Fig. 8 (left plots) as functions of λ.

As expected, we observe that the computational cost does increase along with the size of $\mathcal{Q}$. Therefore, $|\mathcal{Q}|$ acts as a trade-off between the computational cost and the MSE. However, the computational cost of $\widehat{\mathbf{x}}_{\text{online},\mathcal{Q}}$ is still several orders of magnitude lower than the one associated to the online ADMM. Interestingly, computational costs are comparable for $M = 2$ (top left plot) and $M = 5$ (bottom left plot). Note that a warm-up starting strategy for online ADMM only reduces by a factor two the computational cost with respect to the implementation displayed in Fig. 8.

The computational cost of $\widehat{\mathbf{x}}_{\text{online},\mathcal{Q}}$ could still be reduced in two ways. First, one could design the set $\mathcal{Q}$ according to a priori knowledge of components amplitudes (see 5.2). Second, one could also benefit from the separable form of the algorithm and compute solutions $\widehat{\mathbf{x}}^{(q)}$ in parallel for every $q \in \{1, \ldots, |\mathcal{Q}|\}$.

Change-point detection accuracy. The Jaccard index $J(\boldsymbol{\alpha}, \boldsymbol{\beta}) \in [0, 1]$ between any $\boldsymbol{\alpha}$ and $\boldsymbol{\beta} \in [0, 1]^N$ is defined as [18, 22]

$$J(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{\sum_{i=1}^{N} \min(\alpha_i, \beta_i)}{\sum_{\substack{1 \leq i \leq N \\ \alpha_i > 0, \beta_i > 0}}^{N} \frac{\alpha_i + \beta_i}{2} + \sum_{\substack{1 \leq i \leq N \\ \beta_i = 0}} \alpha_i + \sum_{\substack{1 \leq i \leq N \\ \alpha_i = 0}} \beta_i}. \tag{50}$$

It varies from 0, when $\boldsymbol{\alpha} \cap \boldsymbol{\beta} = \emptyset$, up to 1 when $\boldsymbol{\alpha} = \boldsymbol{\beta}$. The Jaccard index is a demanding measure: As an example, if $\boldsymbol{\beta} \in \{0, 1\}^N$ is the truth and if $\boldsymbol{\alpha} \in \{0, 1\}^N$ has correctly identified half non-zero values of $\boldsymbol{\beta}$ but has misidentified the other half, then $J(\boldsymbol{\alpha}, \boldsymbol{\beta}) = 1/3$.

The Jaccard index is used to measure the similarity between change-point locations of $\mathbf{x}$ and those obtained during the computation of $\widehat{\mathbf{x}}_{\text{win},K}$ and $\widehat{\mathbf{x}}_{\text{online},\mathcal{Q}}$. To this end, we consider the change-point indicator vector $\boldsymbol{r} = (r_i)_{1 \leq i \leq N}$ of $\mathbf{x}$ (as well as $\widehat{\boldsymbol{r}}_{\text{win},K}$ and $\widehat{\boldsymbol{r}}_{\text{online},\mathcal{Q}}$ respectively associated to $\widehat{\mathbf{x}}_{\text{win},K}$ and $\widehat{\mathbf{x}}_{\text{online},\mathcal{Q}}$), defined as

$$r_i = \begin{cases} 1, & \text{if } \mathbf{x} \text{ has a change-point at location } i, \\ 0, & \text{otherwise.} \end{cases} \tag{51}$$

In order to incorporate a tolerance level on change-point locations, $\boldsymbol{r}$, $\widehat{\boldsymbol{r}}_{\text{win},K}$ and $\widehat{\boldsymbol{r}}_{\text{online},\mathcal{Q}}$ are first convolved with a Gaussian kernel of size 10 with a standard deviation of 3.

$J(\widehat{\boldsymbol{r}}_{\text{win},K}, \boldsymbol{r})$ and $J(\widehat{\boldsymbol{r}}_{\text{online},\mathcal{Q}}, \boldsymbol{r})$ are averaged over 10 realizations of noise and reported in Fig. 8 (right plots) as functions of λ for different set size $|\mathcal{Q}|$ and window size K.

Performance show that $J(\widehat{\boldsymbol{r}}_{\text{online},\mathcal{Q}}, \boldsymbol{r}) \geq J(\widehat{\boldsymbol{r}}_{\text{win},K}, \boldsymbol{r})$ for almost all λ and $|\mathcal{Q}|$. Therefore, $\widehat{\mathbf{x}}_{\text{online},\mathcal{Q}}$ provides a better online detection of change-points of $\mathbf{x}$. It also show that $J(\widehat{\boldsymbol{r}}_{\text{online},\mathcal{Q}}, \boldsymbol{r})$ does not vary significantly with $|\mathcal{Q}|$ but slightly decreases with M. Indeed, as M increases, the prolongation condition (39) is more likely to be violated, thus leading to more change-points.
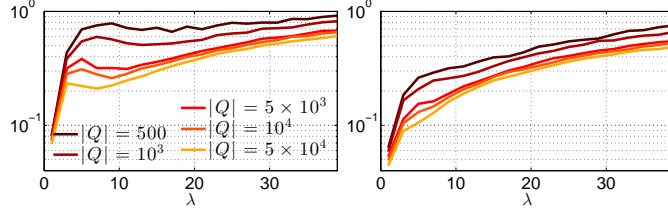
Figure 6: **Estimation performance $\widehat{\mathbf{x}}_{\mathbf{approx},\mathcal{Q}}$ vs $\widehat{\mathbf{x}}$.** $\mathrm{MSE}(\widehat{\mathbf{x}}_{\mathrm{approx},\mathcal{Q}}, \widehat{\mathbf{x}})$ for different $|\mathcal{Q}|$. SNR is set to 4dB (resp. 10dB) on left plot (resp. right plot).
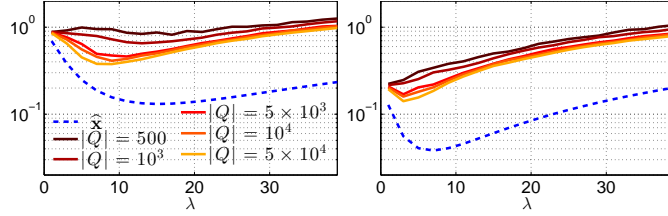

Figure 7: **Estimation performance $\widehat{\mathbf{x}}$ vs. x and $\widehat{\mathbf{x}}_{\mathbf{approx},Q}$ vs. x.** $\mathrm{MSE}(\widehat{\mathbf{x}}, \mathbf{x})$ and $\mathrm{MSE}(\widehat{\mathbf{x}}_{\mathrm{approx},\mathcal{Q}}, \mathbf{x})$ for different $|\mathcal{Q}|$. SNR is set to 4dB (resp. 10dB) on left plot (resp. right plot).

6   Conclusion

In this contribution, we have developed an algorithm which provides an on-the-fly approximate solution to the multivariate total variation minimization problem. Besides a thorough examination of the KKT conditions, the key-step of the algorithm lies in updating and controlling the range of the upper and lower bounds of the dual solution within a tube of radius λ. An on-the-fly derivation is achieved by means of an auxiliary vector $\widehat{\mathbf{z}}$, which needs to be estimated, providing information on the angle of contact with the tube. The latter estimation strongly affects the quality of the solution and the proposed on-the-fly estimation of $\widehat{\mathbf{z}}$ is currently achieved by assigning a value chosen within a predefined set $\mathcal{Q}$. It has been shown that the size of $\mathcal{Q}$ permits to achieve a desired trade-off between the targeted quality of the solution and the application-dependent affordable computational cost. In addition, the proposed method could also be extended to other $\ell_{1,p}$ penalization norms in the right-hand side of (2), for $p > 1$. However one would still face the issue of estimating $\widehat{\mathbf{z}}$ which would have to lie within a ℓ_p ball of radius λ. Under current interest is the investigation of how to estimate $\widehat{\mathbf{z}}$ in the case where the assumption of piece-wise constant behavior is a priori relaxed.

7   Appendix

7.1   Proof of Equation (20)

According to the primal-dual relation (7), for every $m \in \{1, \ldots, M\}$ and $k \in \{1, \ldots, N-1\}$,

$$\widehat{u}_{m,k} = y_{m,k} + u_{m,k-1} - \widehat{x}_{m,k}, \tag{52}$$
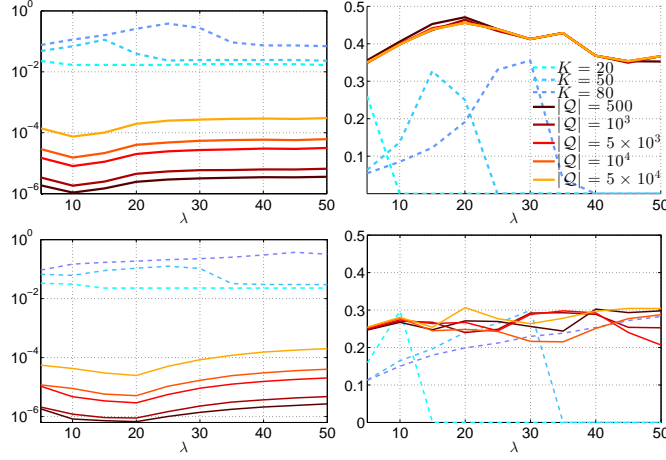
18

Figure 8: **Online Performance.** The proposed solution $\widehat{\mathbf{x}}_{\text{online},\mathcal{Q}}$ is displayed in solid line while the online ADMM solution $\widehat{\mathbf{x}}_{\text{win},K}$ is displayed in dashed line. Performance for $M = 2$ (resp. $M = 5$) are illustrated top (resp. bottom). Left: median computational cost per incoming sample (in seconds). Right: $J(\widehat{\boldsymbol{r}}_{\text{win},K}, \boldsymbol{r})$ and $J(\widehat{\boldsymbol{r}}_{\text{online},\mathcal{Q}}, \boldsymbol{r})$ for different values of $|\mathcal{Q}|$ and K.

and by definition of the lower and upper bounds of $\widehat{x}_{m,k}$ and $\widehat{u}_{m,k}$, we have

$$\underline{u}_{m,k} = y_{m,k} + \widehat{u}_{m,k-1} - \underline{x}_{m,k}, \tag{53}$$

$$\overline{u}_{m,k} = y_{m,k} + \widehat{u}_{m,k-1} - \overline{x}_{m,k}. \tag{54}$$

By subtracting (53) from (52) we obtain

$$\widehat{u}_{m,k} - \underline{u}_{m,k} = \underline{x}_{m,k} - \widehat{x}_{m,k} \tag{55}$$

and, according to (18), $\widehat{u}_{m,k} - \underline{u}_{m,k} \leq 0$. The arguments are similar for proving that $\widehat{u}_{m,k} > \overline{u}_{m,k}$.

7.2   Proof of Equation (27)

For every $m \in \{1, \ldots, M\}$ and $k \in \{k_0, \ldots, N - 2\}$, if

$$\underline{u}_{m,k+1} = \underline{u}_{m,k} + y_{m,k+1} - \underline{x}_{m,k} > +\widehat{z}_{m,k+1}, \tag{56}$$

then updating rules of $\underline{x}_{m,k}$, specified in (23), have under-evaluated its value $\underline{\nu}_m$. To modify the lower bounds $(\underline{x}_{m,j})_{k_0 \leq j \leq k+1}$, on the one hand, we consider the cumulative sum of the observations which, according to (7), leads to

$$\sum_{j=k_0+1}^{k+1} y_{m,j} = u_{m,k+1} - u_{m,k_0} + (k - k_0 + 1)x_{m,k+1}, \tag{57}$$

and thus, if $\underline{u}_{m,k+1} = +\widehat{z}_{m,k+1}$, would lead to

$$\sum_{j=k_0+1}^{k+1} y_{m,j} = \widehat{z}_{m,k+1} - u_{m,k_0} + (k - k_0 + 1)\underline{\nu}_m, \tag{58}$$

19

by definition of $\underline{x}_{m,k+1} = \underline{\nu}_m$. On the other hand, the updating rules (22) and (23) have led to

$$\underline{u}_{m,k+1} = \underline{u}_{m,k_0} + \sum_{j=k_0+1}^{k+1} y_{m,j} - (k - k_0 + 1)\underline{x}_{m,k}. \tag{59}$$

The combinaison of (58) and (59) leads to

$$\underline{\nu}_m = \underline{x}_{m,k} + \frac{-\underline{u}_{m,k_0} + \underline{u}_{m,k+1} - \widehat{z}_{m,k+1} + \widehat{u}_{m,k_0}}{k - k_0 + 1}. \tag{60}$$

Because $\underline{x}_{m,k}$ have been under-evaluated and by definition $\widehat{u}_{m,k_0} \leq \underline{u}_{m,k_0}$, we can propose the following value

$$\underline{\nu}_m = \underline{x}_{m,k} + \frac{\underline{u}_{m,k+1} - \widehat{z}_{m,k+1}}{k - k_0 + 1}, \tag{61}$$

in order to adjust the lower bounds, i.e.,

$$(\forall j \in \{k_0, \ldots, k+1\}) \quad \underline{x}_{m,j} = \underline{\nu}_m. \tag{62}$$

In addition, as a result of $\widehat{u}_{m,k+1} \in [-\widehat{z}_{m,k+1}, +\widehat{z}_{m,k+1}]$ and according to the inequality (20), we set

$$\underline{u}_{m,k+1} = +\widehat{z}_{m,k+1}. \tag{63}$$

References

[1] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo. An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems. *IEEE Trans. Image Process.*, 20(3):681–695, Mar. 2011.

[2] C. Alais, A. Barbero, and J. Dorronsoro. Group fused lasso. In V. Mladenov, P. Koprinkova-Hristova, G. Palm, A. Villa, B. Appollini, and N. Kasabov, editors, *Artificial Neural Networks and Machine Learning, ICANN 2013*, volume 8131 of *Lecture Notes in Computer Science*, pages 66–73. 2013.

[3] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Optimization with sparsity-inducing penalties. *Foundations and Trends in Machine Learning*, 4(1):1–106, 2012.

[4] A. Barbero and S. Sra. Fast Newton-type methods for total variation regularization. In *International Conference on Machine Learning*, pages 313–320, June 2011.

[5] H. H. Bauschke and P. L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces.* Springer, New York, 2011.

[6] K. Bleakley and J. P. Vert. The group fused Lasso for multiple change-point detection. Technical report, Preprint, Hal-00602121, 2011.

[7] J. Cannon, P. A. Krokhmal, R. V. Lenth, and R. Murphey. An algorithm for online detection of temporal changes in operator cognitive state using real-time psychophysiological data. *Biomedical Signal Processing and Control*, 5(3):229–236, 2010.

[8] A. Chambolle. An algorithm for total variation minimization and applications. *J. Math. Imag. Vis.*, 20(1-2):89–97, Jan. 2004.

[9] A. Chambolle, V. Caselles, D. Cremers, M. Novaga, and T. Pock. An introduction to total variation for image analysis. In *Theoretical Foundations and Numerical Methods for Sparse Recovery*, volume 9, pages 263–340. De Gruyter, 2010.

[10] P. L. Combettes and J.-C. Pesquet. A proximal decomposition method for solving convex variational inverse problems. *Inverse Problems*, 24(6):x+27, Dec. 2008.

[11] P. L. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. In H. H. Bauschke, R. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, editors, *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pages 185–212. Springer-Verlag, New York, 2010.

[12] L. Condat. A direct algorithm for 1D total variation denoising. *IEEE Signal Process. Lett.*, 20(11):1054–1057, 2013.

[13] P. Davies and A. Kovac. Local extremes, runs, strings and multiresolution. *Ann. Stat.*, 29(1):1–65, 2001.

[14] N. Dobigeon, J.-Y. Tourneret, and J. D. Scargle. Joint segmentation of multivariate astronomical time series: Bayesian sampling with a hierarchical model. *IEEE Trans. Signal Process.*, 55(2):414–423, Feb. 2007.

[15] M. Godinez, A. Jimenez, R. Ortiz, and M. Pena. On-line fetal heart rate monitor by phonocardiography. *International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 3141–3144, 2003.

[16] A. Gramfort, M. Kowalski, and M. Hämäläinen. Mixed-norm estimates for the M/EEG inverse problem using accelerated gradient methods. *Physics in Medicine and Biology*, 57(7):1937–1961, Mar 2012.

[17] M. Grasmair. The equivalence of the taut string algorithm and BV-regularization. *J. Math. Imag. Vis.*, 27:59–66, 2007.

[18] R. Hamon, P. Borgnat, P. Flandrin, and C. Robardet. Discovering the structure of complex networks by minimizing cyclic bandwidth sum. *arXiv preprint arXiv:1410.6108*, 2014.

[19] Z. Harchaoui and C. Levy-Leduc. Catching change-points with Lasso. In *Proc. Ann. Conf. Neur. Inform. Proc. Syst.*, volume 20, pages 617–624, Cambridge, MA, USA, 2008.

[20] W. Hinterberger, M. Hintermüller, K. Kunisch, M. von Oehsen, and O. Scherzer. Tube methods for BV regularization. *J. Math. Imag. Vis.*, 19(3):219–235, 2003.

[21] H. Hoefling. A path algorithm for the fused Lasso signal approximator. Technical report, Preprint, arXiv:0910.0526, 2009.

[22] P. Jaccard. Distribution de la flore alpine dans le bassin des Dranses et dans quelques régions voisines. *Bulletin de la Société Vaudoise des Sciences Naturelles*, (37):241–272, 1901.

[23] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky. An interior-point method for large-scale ℓ_1-regularized least squares. *J. Mach. Learn. Res.*, 1(4):606–617, 2007.

[24] J. Liu, L. Yuan, and J. Ye. An efficient algorithm for a class of fused Lasso problems. In *Proc. ACM SIGKDD*, pages 323–332, 2010.

[25] C. Louchet and L. Moisan. Total variation as a local filter. *SIAM J. Imaging Sci.*, 4(2):651–694, 2011.

[26] E. Mammen and S. Van de Geer. Locally adaptive regression splines. *Ann. Stat.*, 25(1):1–434, 1997.

[27] J.-C. Pesquet and N. Pustelnik. A parallel inertial proximal optimization method. *Pac. J. Optim.*, 8(2):273–305, Apr. 2012.

[28] F. Picard, S. Robin, M. Lavielle, C. Vaisse, and J.-J. Daudin. A statistical approach for array CGH data analysis. *BMC Bioinformatics*, 6:27, 2005.

[29] A. Rinaldo. Properties and refinements of the fused Lasso. *Ann. Stat.*, 37(5B):2922–2952, 2009.

[30] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Phys.D*, 60(1-4):259–268, Nov. 1992.

[31] G. Steidl, S. Didas, and J. Neumann. Splines in higher order TV regularization. *Int. J. Comp. Vis.*, 2006.

[32] J.-P. Vert and K. Bleakley. Fast detection of multiple change-points shared by many signals using group LARS. In *Proc. Ann. Conf. Neur. Inform. Proc. Syst.*, volume 23, pages 2343–2352, Vancouver, Canada, 2010.

[33] C. Vogel. *Computational Methods for Inverse Problems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2002.

[34] B. Wahlberg, S. Boyd, M. Annergren, and Y. Wang. An ADMM algorithm for a class of total variation regularized estimation problems. In *Preprints of the 16th IFAC Symposium on System Identification*, pages 83–88, 2012. QC 20121112.

[35] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *J. R. Statist. Soc. B*, 68:49–67, 2006.